In the Claims:

Please amend claim 1 as follows:

1. (Amended) I claim a method comprising of a technology element that calculates Multi-Axes Tool Compensation internal to a central mathematical set of algorithms in memory of the CNC controller which ties all of the provided set of commands together as described and shown in the provided flowchart in block diagram form FIG 10. enumerated as Paragraph [0030] titled as Intelligent Database subroutine and Database subroutine which calls and ties to and works together with the group of elements titled the collection of mathematical subroutine elements enumerated as Paragraphs [0031] through [0054] and specifically linked to and shown in FIG 10 of the block diagram as it interacts with the Element titled DbAtr enumerated as paragraph [0043], Element titled DbGet enumerated as paragraph [0044], Element titled DbSet enumerated as paragraph [0045] and Element titled DbSetAtrCur enumerated as paragraph [0046]. To further recite all of the elements, components and steps completely constituting every aspect we further list and explain the technology elements as comprising of:

a. The user setting his or her preferences for the values or amounts to compensate into boxes on an operator screen, such as the example screen in FIG 1 for the boxes labeled tool size, horizontal offset, vertical offset, tool wear, corner radius, bottom angle, side angle and tool length. These interact with the G code program and other values optionally inputted or gathered as variables when the math calculations are performed.

   The user input boxes on FIG 1 specifically are read and stored by the Subroutine Element Form_Load enumerated as [0031] which reads in all data from user input boxes from FIG 1 and stores them into the Database Element as described and enumerated as paragraph [0030]. Further database variables and user settings both public, global and private call, ties to and works together with the database element titled subroutine DbAtr enumerated as [0043] which is a Database element to store geometry properties, error, conditions and positions. This ties to and works with the Element titled Intelligent Database subroutine enumerated as paragraph [0030] and the database element titled Subroutine DbGet enumerated as [0044] which gets Database item coordinate, property and position from random file which works with the Element titled Intelligent Database subroutine enumerated as paragraph [0030] the database element titled Subroutine DbSet enumerated as [0045]. The DbSet Database element sets item coordinate, property and position from random file. The database element titled Subroutine GloRead enumerated as [0032] reads in all global and public data from user input boxes plus any proprietary settings from FIG 1 and stores them into the Database Element as described in and enumerated as paragraph [0030] titled as Intelligent Database subroutine and Database subroutine.

b. The user must repeat the steps in Claim 1.a setting and entering his or her preferences for each tool description. There is no limit to the number of tools, machine types or tool shape combinations to enter.

c. An industry standard G Code program, as in FIG 9, containing tool positions based on non-compensated original part geometry data, interact with the Multi-Axes tool compensation calculations when they are applied. These are the original tool positions that the user supplies in which the calculations are applied. These interact with values provided on the tool parameter screen. For each multi-axes X,Y,Z,A,B,C value entered in the G Code program, the controller will calculate a compensated value based on the amounts entered into the tool parameter screen as in the example screen in FIG 1.

d. A set of optional text entered commands are provided to interact and be directly entered onto the operator screen to override or toggle features on/off and adjust values:

TOOLCOMP OFF	'Turns all compensation off.
TOOLCOMP LEFT	'Comps tool in 2D to the left.
TOOLCOMP RIGHT	'Comps tool in 2D to the right.
TOOLCOMP 3DCOMP	'3D comp based on vector and gouge parameter.
TOOLCOMP 3DADJUSTZ	'3D comp lifts Z axis only but keeps X,Y.
TOOLCOMP 3DOFFSET	'3D parallel offset only - based on vector and 'no gouge parameter.
TOOLCOMP 5AXIS	'5-axis comp based on vector and gouge parameter.
TOOLCOMP LLIMIT45	'Give angle which will specify a gouge to omit tool.

e. A multi-axes tool positioner in a tool holder mounted to a machine's spindle cuts the part as shown in FIG 7 and FIG 8.

f. The process of gathering the user-entered information, preferences, values, amounts, on/off options on the operator screen, as in FIG 1, or as entered by optional text commands, as in Claim 1.d, interact with the original tool positions as provided in the G code program, as in FIG 9, to provide the mathematical variables when processed by a set of described central mathematical routines internal to the CNC Controller as outlined in the DETAILED DESCRIPTION OF THE INVENTION section. The various methods for gathering the information are incidental as to how the central set of math routines that perform these calculations receive them.

Please amend claim 2 as follows:

2. (Amended) I claim a method for Multi-Axes Tool Compensation, specifically pointing out by distinctly claiming the subject matter as what I regard as my invention is an element, which automatically calculates tool gouge avoidance protection internal to the CNC controller's central set of math routine algorithms as shown:

a. Elements and components are depicted in FIG 3 Item 5, FIG 4 DIM "D" Item 6, FIG 5 DIM "E" Item 7 and FIG 6 DIM "R" Item 8.

b. Relationships between the Tool Parameter, Tool Definitions, Machine and Fixture offset elements and how they work together are shown by flowchart in block diagram form, as in FIG 10.

c. We further recite the relationships as outlined in the DETAILED DESCRIPTION OF THE INVENTION in paragraphs enumerated as [0024] through [0028] as the means in which the necessary parameters are gathered from the user and stored using computer variables within the computer's memory as shown in paragraph enumerated as [0030] as the element titled Database subroutine then passed to the subroutines of the technology element as depicted in paragraphs [0022] as the element titled Vector and Matrix subroutine, [0023] as the element titled Gouge subroutine and [0029] as the element titled Central subroutine.

Please amend claim 3 as follows:

3. (Amended) I claim a method for Multi-Axes Tool Compensation, specifically pointing out by distinctly claiming the subject matter as what I regard as my invention is an element which automatically contains algorithms to lift the tool to safe positions or skip the move when necessary by determining if the LLIMIT parameter, as shown in FIG 5 DIM "E" Item 7, is in violation of any surrounding obstacles as determined by a user-defined variable value as enter on the operator screen in FIG 1.

a. This claim is a claim method of calculating tool gouge avoidance and tool protection as outlined in Claim 2.

b. Relationships between the Tool Parameter, Tool Definitions, Machine and Fixture offset elements and how they work together are shown by flowchart in block diagram form, as in FIG 10. As the key objective we make use of a function using the LLIMIT element in the algorithm to be processed by the element titled Gouge subroutine described in paragraph enumerated as [0023]. We further recite the relationships as outlined in the DETAILED DESCRIPTION OF THE INVENTION in paragraphs enumerated as [0001] through [0012] as the means in which the necessary parameters are gathered from the user and stored using computer variables within the computer's memory as shown in paragraph

enumerated as [0030] as the element titled Database subroutine then passed to the subroutines of the technology element as depicted in paragraphs [0022] as the element titled Vector and Matrix subroutine and [0029] as the element titled Central subroutine.

Please amend claim 4 as follows:

4. (Amended) I claim a method specifically pointing out by distinctly claiming the subject matter as what I regard as my invention is an element to redefine, replace and override the tool position coordinates when the tool characteristics change:

a. Relationships between the user-definable command method elements and how they work together are shown in the flowchart in block diagram form, as in FIG 10.

b. As the key objective we use a method of user-definable command elements that override, replace and redefine the variables gathered from the user and stored using computer variables within the computer's memory as shown in paragraph enumerated as [0030] as the element titled Database subroutine.

We list and recite the elements of the user-definable commands as shown in paragraphs enumerated as [0015] through [0021] of the DETAILED DESCRIPTION OF THE INVENTION as:

TOOLCOMP OFF
TOOLCOMP LEFT
TOOLCOMP RIGHT
TOOLCOMP 3DCOMP
TOOLCOMP 3DADJUSTZ
TOOLCOMP 3DOFFSET
TOOLCOMP 5AXIS
TOOLCOMP LLIMIT45

which are further recited to show the relationships of how the user-definable command set overrides, replaces and redefines the user input as outlined in FIG 1 on the operator screen and described in the DETAILED DESCRIPTION OF THE INVENTION in paragraphs enumerated as [0001] through [0012].

Please cancel claim 5.

Please cancel claim 6.

Please cancel claim 7.

Please cancel claim 8.

Please amend claim 9 as follows:

9. (Amended) I claim an algorithm element expands the intelligence of each calculation for compensated tool positions based on an artificial intelligence algorithm element.

a. The artificial intelligence algorithm element is actually a live, real-time, ever-changing database in the machine's memory that remembers by learning from what the machine can and cannot do. The database is a storage of events, variables as an internal list of conditions and positions kept in standard random access memory as outlined by the various variables used by the central set of math algorithms.

b. Specifically pointing out by distinctly claiming the subject matter as what I regard as my invention is an element for which we recite as a method to store events, conditions, positions and errors into computer variables within the computer's memory as shown in paragraph enumerated as [0030] as the element titled Database Subroutine.

c. Relationships between the intelligent database element that stores the events, conditions, positions and errors into computer variables and how they work together are shown in the flowchart in block diagram form, as shown in FIG 10.